

# Meta Data to Support Context Aware Mobile Applications

Dan Chalmers

Naranker Dulay

Morris Sloman

Department of Computing, Imperial College London, London SW7 2AZ, U.K.

E-mail: d.chalmers, n.dulay, m.sloman@imperial.ac.uk

## Abstract

*The greater sophistication and availability of mobile computing devices is enabling the application of computing to new tasks and applications to be experienced in a wide range of contexts. To facilitate such applications it is necessary to adapt the applications and the data they display to operate within the limitations of the context of use, take advantage of the strengths of the devices and best support the users in their tasks. A part of the solution to this issue is the description of the data available to the applications.*

*Meta data may describe the structure and properties of information and data. We achieve this by separating the semantic elements of the information from the syntactic variants which realise them in order to form documents.*

*In this paper we describe a new meta data model which can encode this description. We provide example meta data and illustrate how this may lead to better application usability than current simplistic or constrained approaches, through contextual mediation.*

## 1 Introduction

The power and range of available mobile devices is becoming greater. As these devices become more popular and computing use more ubiquitous a wider range of data will be required by users to fulfil their tasks. The structure of data will become richer – the current move from flat file formats with single purpose design to interlinked XML documents with well identified sub-elements representing different information will continue and support rich information applications. The data-intensive nature of some data formats could, however, overwhelm limited mobile devices.

The variation in data forms and the degree to which data are structured also supports wide ranging variations in the presentation of information, in order to work within the limits of these systems and also to best support the needs of users. In some cases the transformation of data will be dramatic and in others a slight change in the context of use will result in a subtle change in the presentation of data. We

call this adaptation of data and its presentation to the context of use *contextual mediation* [3, 4]. In order to support contextual mediation it is necessary to provide meta data, to enable the selection of the most appropriate forms and the omission of unwanted data. Examples of the situations meta data should support include: devices with limited ability to display certain formats; the need for deadlines in ensuring responsive applications; the ability to select a combination of data which will remain readable on a small screen; use of data that can be processed for speech output; and also the selection of data which is most useful to the user from a rich information space. We use meta data to describe the structure and alternative presentations of information, but not to guide the adaptation explicitly, unlike [7, 8, 9, 11] among others. In this paper we shall describe our meta data model and how it may be applied to different applications to enable contextual mediation.

We have outlined here the need for meta data in contextual mediation. In the following sections we shall discuss the literature, present our meta data model and discuss implementation issues we have encountered. We now conclude our introduction with a summary of the requirements we place on the meta data model:

- A description of the logical structure of *documents*, so that highly structured data may be presented for mediation as a single request resulting in a set of *elements* to be considered in combination. The combination of data to form documents is missing in such approaches as HTTP and its extensions.
- A description of multiple data *variants* which may realise each element, building on [10]. These offer a range of presentations of information.
- A description of data properties, to enable resource management and allow selection to reflect context.
- A description of data semantics, to enable differential treatment of data elements reflecting user preferences. This aspect of data is often ignored by device-centric forms of context awareness.

- Language(s) to describe structure, properties and semantic types, which are extensible to support developing application classes. We illustrate this using two exemplar applications.
- That the model and languages do not encode how to manipulate data, leaving the response to context diverse, extensible and user centric.

## 2 Background

### 2.1 Structure and Semantics

The treatment of the semantic structure of data in context awareness is limited. In many systems, e.g. [15] there is no provision for considering the overall effect of the various elements of a structured document, each being treated independently. In others, e.g. [9] the structure is well defined but the range of adaptation is highly constrained. Similarly, the use of semantic information to support adaptation of information applications to context is not widespread. [8, 16] both describe systems for transcoding elements of web pages. Their system includes analysis of image content, to differentiate between adverts, textures, bullets, lines, maps, logos, navigation buttons and content. They then describe the use of these types in selecting transcoding to apply, given resource constraints such as display capability and bandwidth. [16] describes a technique for extracting purpose from images automatically. This and similar techniques may be used to aid meta data generation where authors do not wish to provide detailed type information.

### 2.2 Variants and Quality Information

In contrast to the treatment of structure and semantics, much work has been undertaken on the adaptation of data to suit the context of use. Some work chooses to describe quality of media in meta data, such as [18], this however has the failing that it assumes that all users will perceive the quality in the same way. Where the quality is determined by the context in which the media are used this is not the case. Similarly, the results of a query may be ranked by a search engine and some rating given. Whether this ranking matches that a user would give depends on many factors, such as: the search engine used, the query given and any other data on the user used in the query.

Unlike many other approaches [7, 8, 11, 13] no quality rating is provided by the authors of the data or meta data in our approach. The meta data provides only an objective description of the data. The content provider's loss of control is balanced by their ability to specify the type of data represented (which is often the underlying theme in justifications for abstract quality ratings by the provider) and to limit the

variants provided. The benefit is in enabling users of diverse devices with diverse interests to access data for a small overhead in the data provisioning process. The effort required to produce meta-data is significantly lower than that required to hand-tailor data.

One example of previous work which simply describes data and includes the notion of at least somewhat structured multimedia documents is [17, 10], which describes documents comprised from mono-media elements which are available in multiple variants. The variants are described according to their attributes with quality determined by comparison with requirements, as in our system. The media elements have a description and spatial and temporal organisation attributes, but they seem intended as place-holders for data objects rather than a description of the semantic content of the data. Their approach is based on a news on demand application, using streamed video as a main source of data. The data used is explicitly voluminous and unstructured and so the variants offered are coarse grained.

### 2.3 Context Based Data Mediation

The W3C Composite Capability Preferences Profile (CC/PP) [15] notes that mechanisms such as accept headers (in HTTP) and ALT tags (in HTML) are somewhat limited and described the application of the Resource Description Framework (RDF) [12] to describe user preferences and device capabilities in a general content negotiation solution. The IETF CONtent NEGotiation (CONNEX) group's work, described in RFC 2533 [11], gives a system for describing preferences for media based on attributes of the media. The CONNEG group have addressed similar issues to the CC/PP group and there are efforts to ensure that the two systems can inter-operate. The descriptions include support for a detailed description of the hardware platform, software capabilities and preferences for languages and security. While this addresses the need for transmittable preferences we shall have to look elsewhere for techniques to describe the data.

W3C's Synchronized Multimedia Integration Language (SMIL) [9] and the CWI Multimedia Interchange Format (CMIF) GRaphical INTERface for Creating and Playing SMIL (GRiNS) editor and player [2] describe temporal and spatial behaviour of a presentation and have similar constructs for describing multiple variants of media elements in a structured presentation. The selection may be subject to a test on various parameters of the system in SMIL, such as bit rate, language and screen size. The selection is encoded with a switch statement, the first acceptable version of the element being selected. CMIF focuses on providing alternative content for various contexts or user abilities, as defined by the author. This system builds on SMIL to offer groupings of content for different users, e.g. due to the

language they speak.

These systems meet our desire for a system which enables selection over whole documents and provide support for the context. However, in both cases the selection of alternatives is limited by the authored selection support and the author's definition of the conditions under which the variants will be appropriate. These techniques may be suitable for carefully authored multimedia documents, where the presentation is subject to careful production controls, however these techniques are less appropriate for presentations where the data are used in evolving or unpredictable ways, or in contexts which were not planned for.

In [8] a mechanism of external annotation (metadata) which indicates properties of elements of a web document in order to facilitate transcoding is presented. Their system supports the description of alternative representations of documents or their elements and a description of hints on splitting a web page into multiple pages using importance values. They support the description of roles of elements within documents such as: decoration, content, advertisement and the description of an importance value from the author. Transcoding to meet the needs of the client can be performed with knowledge of this information, thus enabling more aggressive transcoding of less important data. The ability to split pages for use of a small screen and to provide effective incremental loading is also useful. While we do not believe that the importance value is always valid the provision of semantic information about elements of a document, as well as properties of data, is important. A mechanism which supports this beyond web pages would clearly be useful in adapting to contextual needs.

A system for annotating web pages (including images etc. referenced within those pages) with semantic information, with the intention of facilitating automatic transcoding of the page is described in [14]. Transcoding proxies maintain user profiles and perform appropriate transcoding according to these, with the semantics of the elements taken into account. The content adaptation is therefore dynamic and the data the user receives may be the product of multiple transcoders. Text is annotated with a semantic description of the information, including the grammatical structure of the sentences. Summaries of text may be produced to a specified size. Image transcoding may modify size, colour depth or compression ratio. In their system the variants of the data are provided on demand, to specification. Our approach provides for existing variants to be described and reused, thus saving effort – particularly in the case of text summarisation where substantial meta data is used to describe grammatical structures and significant per-user processing would be required to provide per-user variants. The structural description our meta data provides is at a higher level and we have shown its application to high level adaptation to context across application domains, while their ap-

proach is focused on providing for transcoding on the web.

## 2.4 Meta Data Standards

There is also an emerging range of standard meta data formats based on XML. Of particular interest is the Dublin Core [1] and OpenGIS [5] work. However, the focus of these standards tends to be on meta data to aid complex searches across data – describing the meaning in the data. The description of the structure of data and the attributes of the data is not their main purpose. The OpenGIS standard however has put a lot of work into vocabularies and syntax to describe geographical entities and we may at least partially reuse these definitions in future work on adapting maps.

RDF [12] is a generalised syntax for meta data and may also be applied to our future work. There is very little description embodied within the RDF standard, it essentially allows for the definition of resources and properties of those objects, which allows the description of relationships between resources. Our approach to meta data could be represented in this way. The meta data presented here is shorter and clearer than that which one may expect from RDF, so more suited to describing our model. It is the concepts in the model rather than the XML which is our focus here – and these could be translated into RDF.

## 2.5 Summary

While there has been substantial work on adapting individual media to context, a system which provides the user control over the selection of complex data in response to context is missing. Describing these data is an important part of the solution to this issue, which we address here. This meta data must provide for a rich context-awareness, where the needs and limits arising will evolve over time – reflecting developing uses of data in the long term and resource variation and local conditions in the short term. The resulting effects may be both substantial and subtle. This flexibility requires a rich description of data which does not impose expectations of use. Our approach differs from that in the literature through the combination of: structural, semantic and data property descriptions with an extensible model allowing its application to different application domains.

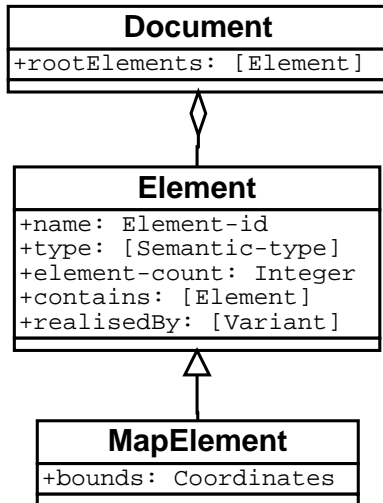
## 3 Semantic Structure

In this section we describe our model of the logical structure of data and the description of semantic types in this logical structure. The description of structure enables mediation to consider documents formed from structured data and adapt different elements of the document differently.

The description of semantic types enables the differentiation of one element from another and identification of similarity. This differentiation allows contextual mediation to identify which elements are most important and so provide a sensible choice of adaptation.

### 3.1 A Model: Semantic Elements of Data

We shall now define the concepts we use in the handling of the semantic structure of data. See figure 1 for an illustration of the classes we use to represent these concepts.



**Figure 1. Meta Data Model (Document and Element Classes for Map Data)**

A *document* is a unit of presentation for structured multimedia information, such as a web page or map. A document is considered to be a collection of one or more elements.

An *element* is a logical unit which fulfils a specific role in a document. In a map example this may correspond to a feature, e.g., the M1 motorway, or river Thames. Elements are described by a *type* which identifies its semantic content, e.g., road or river. Multiple types are allowed to describe complex elements. An element may be represented by one of a choice of *variants*, each providing a different representation in data of that logical element. We shall return to these in the next section. An element may contain other elements, thus describing a logical structure to the information, e.g., to add labels to the representation of a road, or a picture in a web page contained by the HTML which refers to it. Being contained, these elements rely on the containing element to be present in order that they may be displayed. A set of root elements are defined in the document object, forming the roots of the trees of elements. Elements may also be aggregated into groups, e.g. minor roads in an area, rather than describing all elements individually. In this

case the *element-count* parameter will reflect the number of contained elements.

The Element class be extended to represent domain specific concepts. These may have *attributes*, such as the area covered by a map feature as illustrated by the MapElement class.

### 3.2 Semantic Types

The problem of an effective description language requires a *domain* of reference and a published and well understood set of commonly used terms within this. There are many subtleties to the issue of naming, such as disambiguation and cultural terms of reference, which have been the subject of much research. For the purposes of this work we assume that semantic *types* can be described using a system of common ontologies providing typing for domains. Naming within a domain is assumed to be unambiguous.

We have used an ad-hoc type system for map data, drawing from feature types defined in our source data sets. In future typing may be able to draw from the large body of work existing on ontologies. For instance, below is an extract from our type definitions for roads within the domain of maps. Having identified the domain a type of road is defined. Sub-types of road include motorway, a-road etc.

```

<type-domain name="map">
  <type name="road">
    <type name="motorway"/>
    <type name="a-road"/>
    <type name="b-road"/>
    <type name="minor-road"/>
    <type name="under-construction"/>
  </type>
</type-domain>
  
```

The use of domains allows parallel ontologies to be developed independently. Type names need only be unique within a domain. Hence “table” can take different meanings in a type domain for building plans and for web page descriptions, etc.

There is also a need for flexible precision, e.g. some maps describe woodland as simply “woodland”, others may distinguish deciduous and evergreen, others may describe in some detail the types of trees, their age and other data. The map which simply gives woodland as a description should have meta data elements with a type of woodland, not given some spurious accuracy. Our meta data uses a hierarchical type system and so can encode this.

### 3.3 Case Study: Mediation of Maps

One application we have developed using our meta data provides mediated maps. The potential for information

overload in map data in a mobile scenario is high: consider how the detail shown on a small scale paper map would look if shrunk and digitised for a PDA display, then add hyperlinks to shops and tourist attractions, traffic information etc. The problem may be further compounded by the variation in context in which the map is used: vibration or poor lighting may make small details harder to read; other activities such as driving, walking or holding a conversation may limit the attention which can be paid to the map; different devices may have different size, resolution or colour depth in their screens. In order to adapt maps effectively for use in mobile and ubiquitous computing requires that sophisticated choices be made about the data displayed. These should be based on a good description of the available data.

In this application a document is the map for an area as described in a single request to the server. Elements may be single features, e.g. a road; parts of a feature, e.g. the section of a road between two points; or groups of elements which are not described individually, e.g. minor roads in an area which are selected or omitted together. Elements may have structure, e.g. road names and traffic information may depend on the road they describe.

The adaptation of maps may take various forms: zooming in or out; selective display of classes of feature; and offering features at different levels of detail being the most common. Zooming enables clarity by limiting the area covered and changing the map scale, but is limited by the need to view features in the context of their surroundings for most practical uses. Selective display of feature classes offers clarity by removing less important features which may obscure other detail, allowing useful data to be presented with greater separation and/or larger symbols. Again, there are limits to its applicability – many classes of feature, such as major roads, are useful in a wide range of circumstances in order to provide context, even if not directly used. Adjusting the detail presented may offer the simplicity of a more schematic map and is also used to reduce the data volume associated with a feature. This is often combined with zooming in traditional maps.

Vector map data lends itself to partial presentation and processing of different elements. Our meta data's description of the semantic structure of data enabled us to prioritise data according to its type, e.g. roads, rivers, buildings, etc. In this way mode of transport affects the priority of representations of roads, railways and footpaths. Speed affects inclusion of distant or small features. Task (work, leisure etc.) affects inclusion of tourist symbols. Note that the selection of data to display considers the combined effect of all the data in the selection, rather than processing data on an element-by-element basis.

Contextual mediation of a map is illustrated in figure 2. The top map shows the complexity resulting from no mediation. The mediated maps shown are for navigating in a

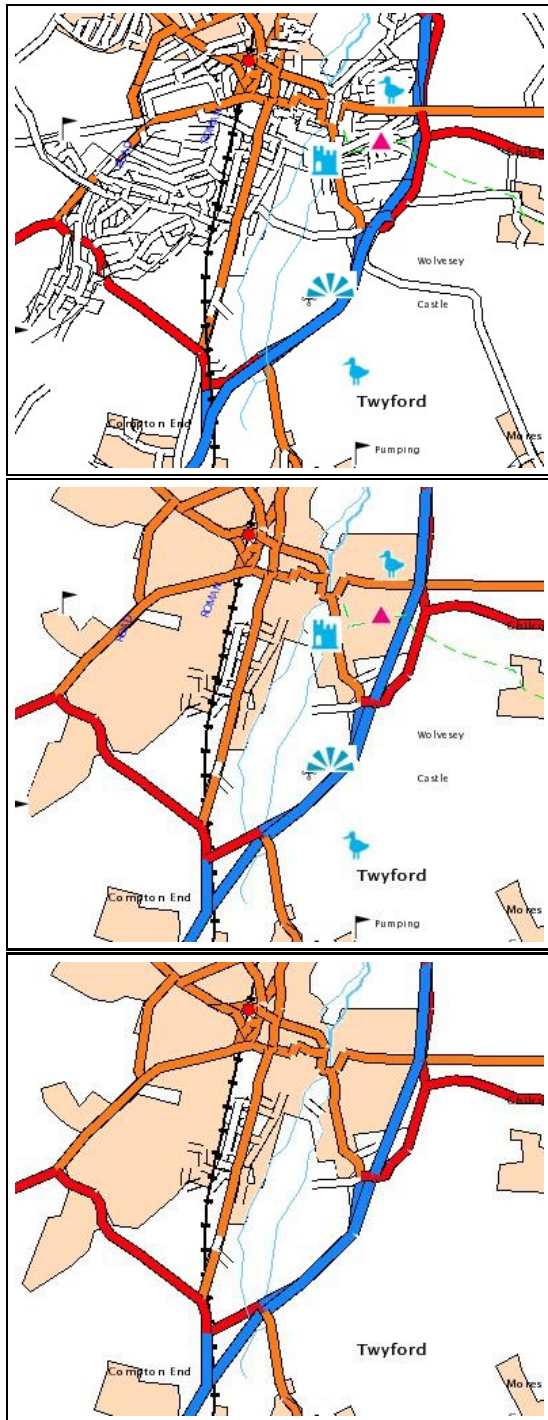
car at 30 to 40mph, hence the concentration on major roads and the omission of minor roads away from the immediate vicinity (the centre). The data was loaded within a 25s deadline over a simulated GPRS network. The difference between the tourist and worker's maps is subtle, but helps the worker to scan the map without distraction. Where features have hyperlinks from them, e.g. describing tourist attractions or access to delivery points, the ease of selecting links also improves. The selection mechanism is described in more detail in [3, 4].

We illustrate in figure 3 the elements in our meta data to describe a motorway and the resulting objects once parsed. We have adjusted the XML to make it more readable, in practice some tags are abbreviated and ids are less human-friendly. This example shows several features of our meta data:

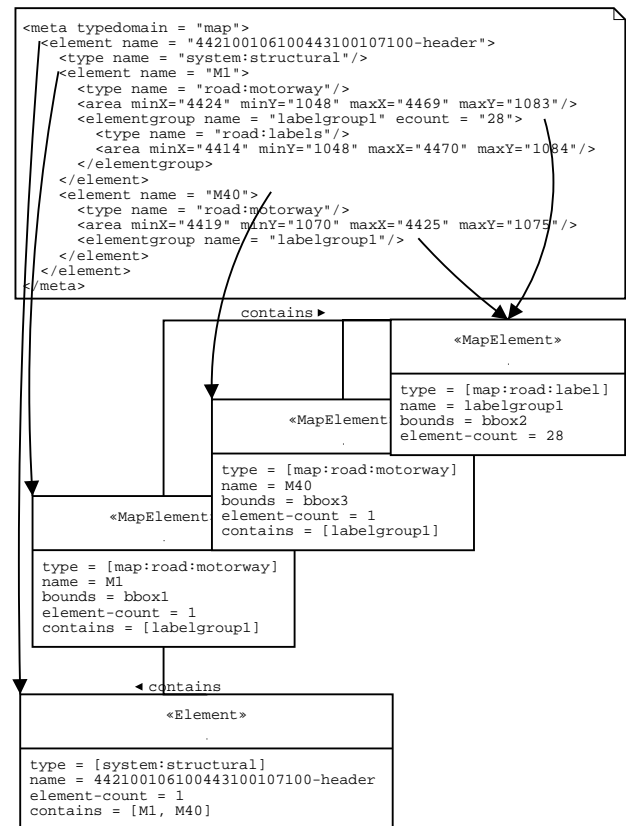
- The meta data opens with a `<meta>` tag, which identifies a default `type-domain` for the `type` definitions in the following elements.
- A containing element for the response to a request, with a unique identifier for that request. The containing element is identified as being a `structural` element rather than conveying information.
- An element describing the "M1". The element's `type` is identified as `motorway`, a subtype of `road` in the domain.
- The `area` defined is the bounding box around the element. This is an extension to the element syntax for use within the map domain, and is interpreted by a map-aware extension of the parser class and held in the `MapElement` subclass.
- A sub element to the motorway is an `element` group of text labels. This may be a generic label definition covering several parts of the M1 which are represented by the 28 grouped elements to reduce meta data overhead.
- The "M40" element also contains the sub-element of the labels. The full definition need not be repeated, so its definition is limited to the `name` field.

While we do not illustrate it here, map elements may have multiple semantic types. For instance, building may contain within it a range of shops, offices and living accommodation, with a wireless phone base-station on the roof. The building might be described simply as a building, which is true but represents a significant loss in precision. To describe the building as having multiple functions would be more useful in many situations.

The location of the element (from its bounding box) is a semantic property which allows selection beyond the simple



**Figure 2. Illustration of Contextual Mediation: Unmediated Map (all features included); Mediated for a tourist in a car; Mediated for a worker in a car (similar, but note omission of tourist features, footpaths etc.)**



**Figure 3. Meta Data Illustration (Element Definitions) and Element Objects Resulting**

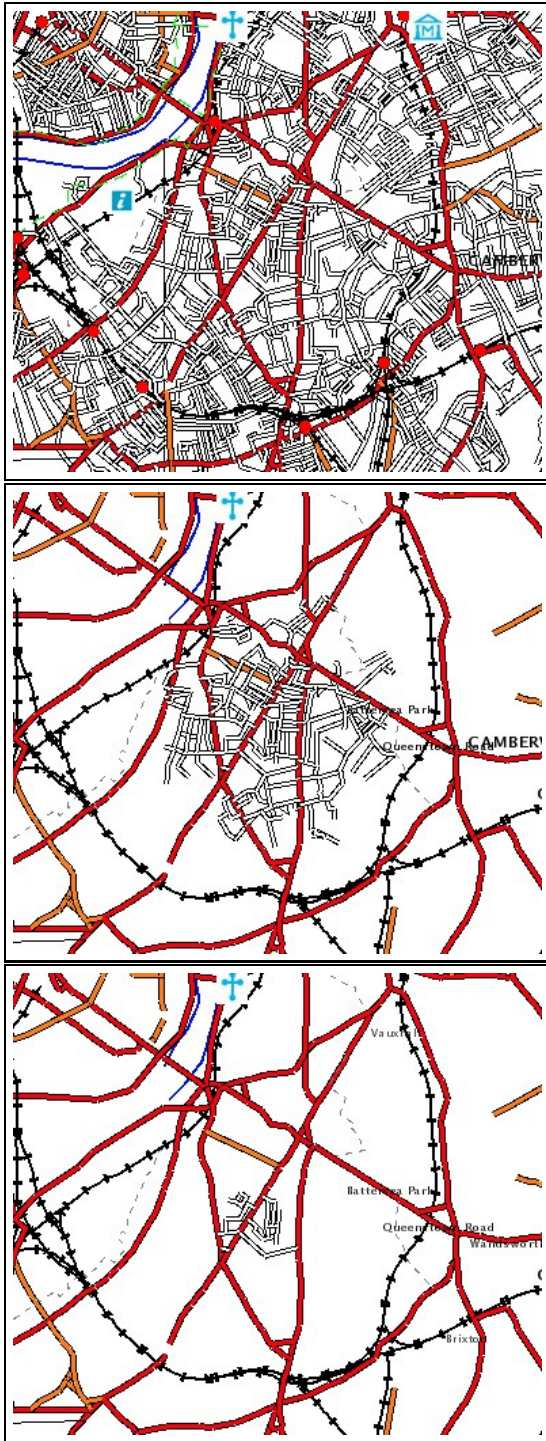
layers approach familiar to GIS users. Selection of greater detail around the focus of interest and a simple context giving map further away allows data volume (and so download time) and screen clutter to be managed in a highly effective way, as figure 4 shows. Each of the maps shown will be appropriate in different contexts – none is ideal in all situations.

Another use for meta data is that it allows the indication of missing data. For instance, if a group of roads was omitted from a map in order to meet a deadline, and roads were defined as sufficiently important, then that area (defined by the bounding box in the meta data) may be given a shading which indicates missing data rather an absence of features. The user may then be able to request a new loading of data to complete that area of the map if the missing data is along their route.

### 3.4 Case Study: Describing Structure for Web Documents

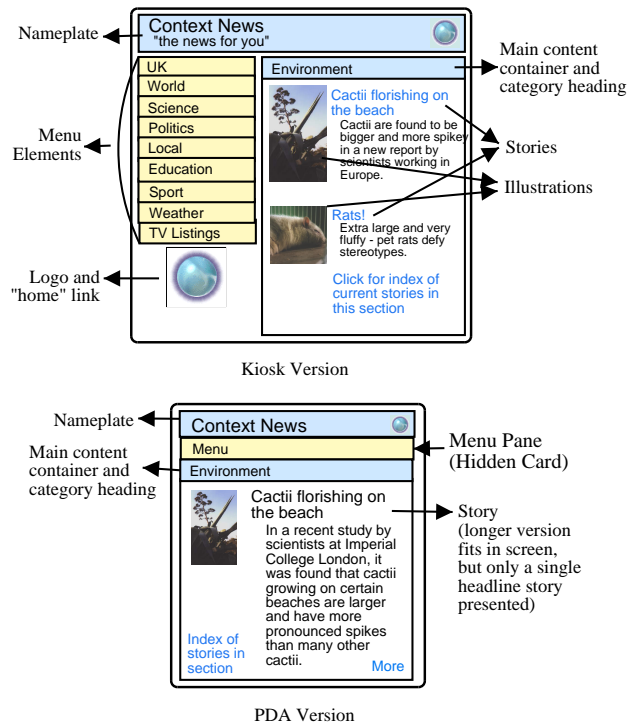
Multimedia data are widely used to represent information and provide an interface to information applications.





**Figure 4. Semantic Attributes Are More Than Types: Here We Illustrate Omission of Progressive Detail By Selecting According To Distance**

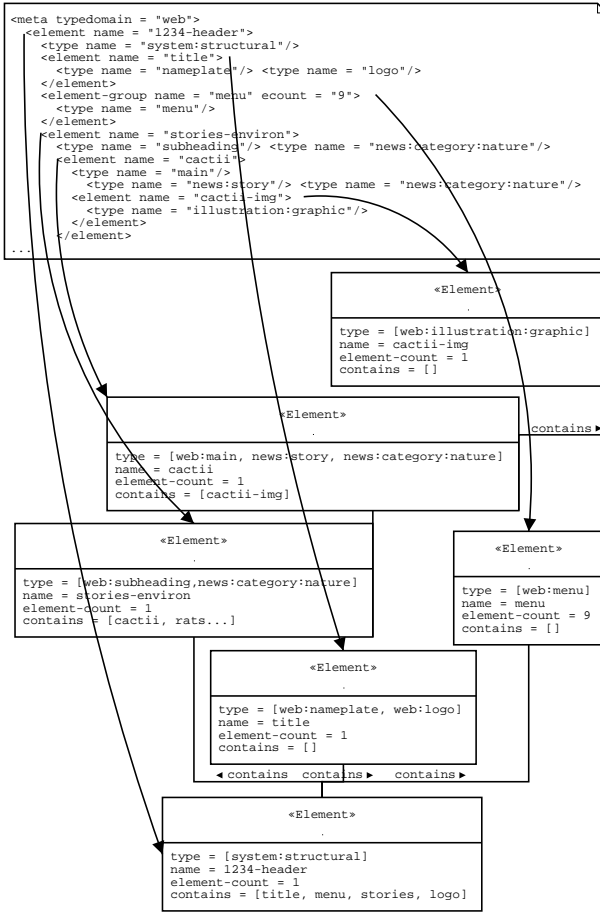
HTML web pages form the basis of many of these. In this case semantic structure remains important, although often coarser grained than in the map data described above. We show an example page in figure 5, where we have a news web site, which has a nameplate, menu, category heading, logo and a section of main content illustrated with images. In figure 6 we illustrate part of the meta data and element objects arising from this web page.



**Figure 5. Alternative Presentations of a Web Site and Illustration of Structural Elements**

The structure of a web page at a semantic level will treat the presence of banners, navigation bars, design graphics, "body content" and illustrations. By separating these elements their representations can be treated differently. In figure 5 we show a small screen where the menu is hidden and can pop forward when a control is activated. The stories in the main content area are presented in different lengths, different size fonts and with different illustrations, depending on screen size, download deadline and other aspects of the context. We see in the meta data example that the page has several layers of containment of elements, reflecting the structure of the information.

Note also in figure 6 the use of three type domains: web as the default domain, system and news. The web domain provides for identification of elements as taking generic roles within a web site, such as nameplate, menu, subheading, main (for main body content, gen-



**Figure 6. Structural Representation of a Web Site**

erally deferring to a subject type system for further classification) and `illustration:graphic` (as opposed to `illustration:infographic:piechart` for instance). The identification of the roles of elements within a page will be used to direct variant selection according to the importance of the element and also to inform context sensitive presentation, allowing the rendering engine to choose appropriate fonts and scaling to balance its drawing of the data selected according to its importance. The system domain is used as before to indicate a structural element for the request. The news type domain is applied in the identification of the `cacti` element as `news:story` and `news:category:nature` and also to the `stories-enviro` element (providing the category heading) as `news:category:nature`. This multi-domain approach allows the generic web structure type definitions and the context-driven response to the type to be managed separately to the type definitions and mediation specifications for news information. Other ontologies may

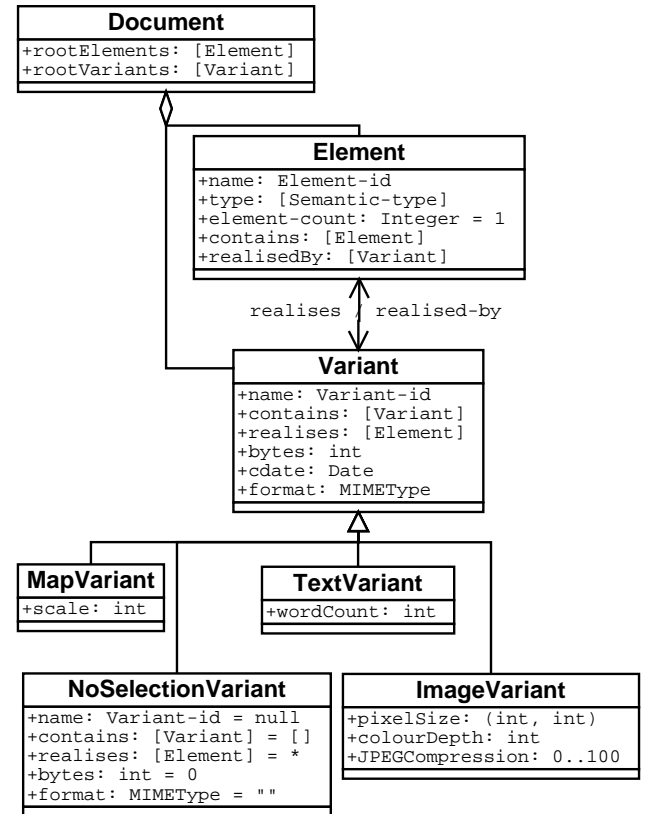
be substituted as long as both server and mediator have the type specification.

## 4 Variants Realising Elements

A description of structure enables mediation to take into account the relationships between media elements and the overall effect of multiple elements. However in order to accurately respond to resource limits and many types of user preference a description of the data realising individual elements is required. We discuss these next.

### 4.1 A Model: Syntactic Variants of Data

As we said before, an element may be represented by one of a choice of *variants*, each providing a different representation in data of that logical element. See figure 7 for an illustration of the classes we use to represent these concepts.



**Figure 7. Meta Data Model (Variant Definitions and Relationship to Elements)**

A variant may contain other variants indicating a dependency in the data which is required for realising the contained variant, e.g. data which requires but may be sepa-



rated from its header. A set of `rootVariants` is defined in the document to provide a starting point for the selection process, as for `rootElements`.

In addition to their structure and relationship to elements, variants must describe their properties to be useful. Various properties are generally applicable: size in bytes, MIME type (format), last modification date etc. Others are specific to the data format (in a similar manner to type domain specific element properties). For maps, scale is a meaningful property. For images, resolution and colour depth would be used. For text, language and word count may be described. The MIME type is used to identify the sub-class of variant which is appropriate in each case.

## 4.2 Case Study: Variants in the Web

The syntactic structure of web data is highly varied and the range of possible presentations of the information may have a substantial impact on the user's experience and on system performance. Much of the prior work on adaptation of data to context has taken the form of *transcoding* – modification of the size of images, encoding format of text, summarisation of text etc. In most cases in the literature this transcoding has taken the form of fixed-behaviour proxy-based translation or selection of pre-generated alternatives, e.g. [6]. These standard adaptations are often designed to reflect device capabilities, such as screen size or capability to render a format.

While fixed or uniform translations are suitable solutions to the problem of device limitations, which tend to remain stable, they are a somewhat blunt tool to apply in adapting to a richer and more dynamic definition of context. Activities, such as driving, may cause summarisation of text to be applied. Speed and proximity to other cars may adjust the degree of summarisation and trigger simplified variants of drawings to be selected. When the hands are occupied using tools, the application interface may be simplified for use with eye-tracking or voice input driven controls. If an augmented reality system is being used then in some cases a clear field of view on a subject of interest may be required, reducing the number of elements displayed or the size of their representation, so that they remain in the periphery of the display.

The representation of elements is varied in these scenarios to enable the most suitable presentation of data for the context, as described above. However, in many cases the end-system in mobile or ubiquitous computing scenarios has neither the processing power, memory or network bandwidth to process data in its original form in order to arrive at the most appropriate form. This task can be fulfilled by the server offering pre-generated variants, proxies offering standard translations or custom variants generated to a specification. In order to be effective the contextual medi-

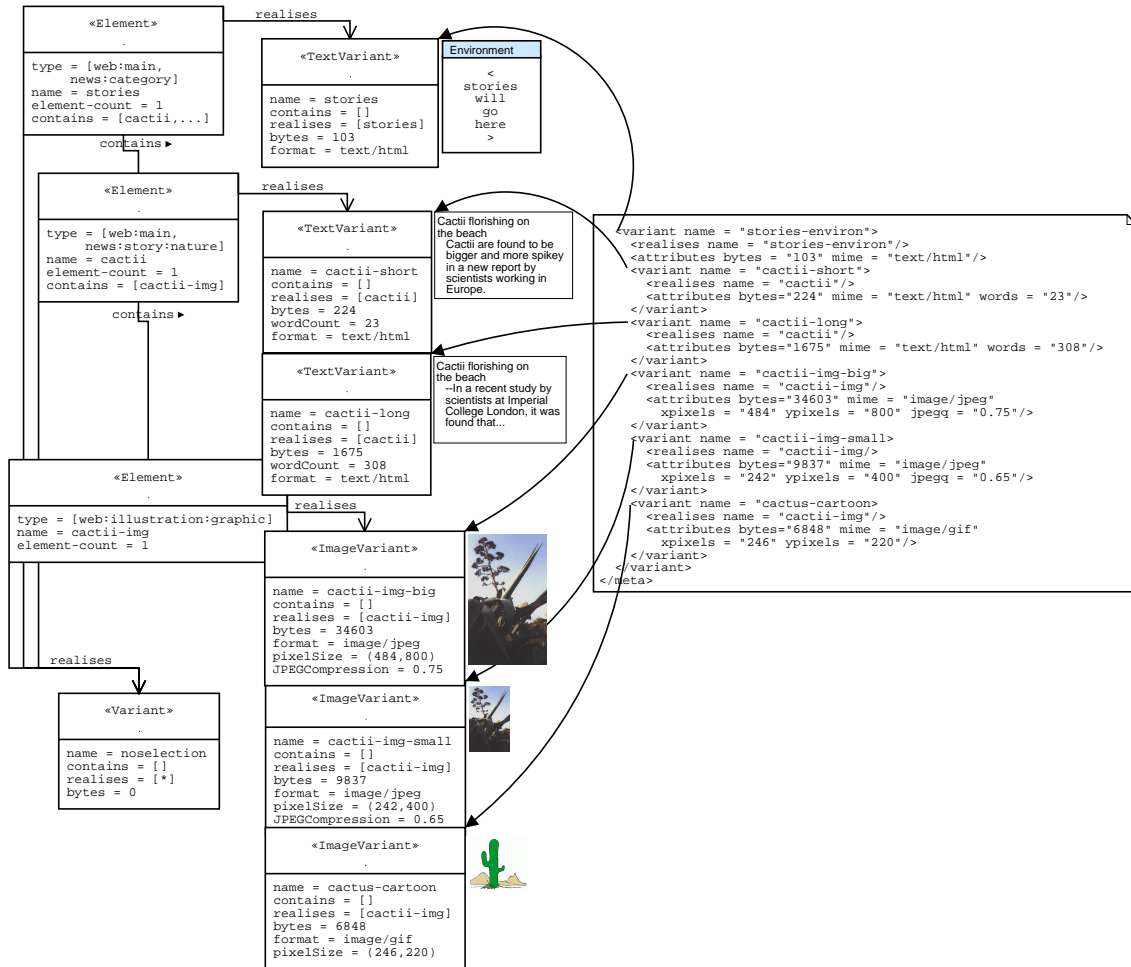
ation system needs to be presented with the available alternatives. In each case the source of the transformed variant can describe the new variant's properties (either exactly or by estimation depending on whether the transformed variant exists or is to be generated on-the-fly).

A tree of dependant variants is described, such as a web frame set, containing HTML frames, images etc. Each of these variants describes the elements which it realises. For each element a "no selection" variant is also assumed to exist, allowing the choice of omitting the element completely. If a variant is not selected then its dependant variants are also not selected. There may be more than one `rootVariant` in the document, which allows a frameset (or other structural container) to be bypassed by including just the story, but not allowing *only* the peripheral elements to be realised. Multiple elements may be realised by a variant but an element will be realised by exactly one variant (which may be the `NoSelection` variant).

Again, we consider the example news site in figure 5. It is likely that various variants of a story will be produced in the normal production process: headline text, a full text story, video and text transcripts from TV bulletins, audio and text transcripts from radio programmes etc. Proxies may then use these sources to generate variants based on summarisation, format transcoding, adding of hyperlinks, language translation etc. Each of these classes of source may also be found in various styles: lightweight presentations, longer in-depth presentations, children's versions etc. While some of these variants may be selected according to a long-term personal or device based preferences in many cases it is also likely that time pressures may affect the detail required in some contexts and that activities being engaged in and social context will affect the mode of presentation, particularly with respect to the choice between text, audio and video presentation.

We illustrate our meta data for defining variants, and the objects resulting from the parser in figure 8. The meta data shown is assumed to continue from that in figure 6, illustrating a web site's structure. Again, we have simplified the syntax and naming slightly for readability and brevity. We shall now describe what this meta data describes:

- A variant `stories` is defined. This realises the `stories` element, which provides the category heading and contains the subsequent variants (note its closing `<\variant>` near the end). The attributes of this variant: its size and MIME type are given.
- The next two variants are alternative forms of the `cacti` story – long and short. Their word length and data size being different. Each of these may be appropriate depending on deadline / network bandwidth, screen size, personal preference etc.
- The next three variants are alternative representations



**Figure 8. Meta Data Illustration (Variant Definitions) and Variant Objects Resulting**

of the graphic which illustrates the story. We have left the differentiation of the photos and cartoon variants for future work.

## 5 Implementation

In this section we present a commentary on the implementation of our map server, its meta data generation and results from testing.

### 5.1 The Map Server

The Ordnance Survey generously made a large body of vector map data in NTF format (BS 7567) to us, providing UK-wide coverage in two different data-sets. Other formats are available and the techniques described here are generally applicable although implementation details would differ. Our map server parses the NTF data files on start-up

and stores the data in element and variant objects. Lines are further processed to remove detail from the sequence of points, in order to offer higher scale variants. The elements are stored in an RTree to enable efficient look-up. Requests can be for meta data or map data, we describe each below.

A meta data request defines an area of interest and returns a description in XML of the elements and variants in that area. Variants of features are described as being contained by a variant of the header data from the source. More than one header variant may be included in any given request. The header includes origin coordinates, from which feature data describes an offset. The element and variant names are generated from the object hashes to ensure uniqueness, even where multiple data sources might use the same feature codes within the data.

The meta data offered is stored in a hash, to save a more complex search for requested variants when a data request is made. The hash key is the name of the header element,

which is derived from the area the data requested is for. This key is used again by the client when data is requested. Requests are periodically removed from the hash, to allow the server to garbage collect the objects and control memory use.

A data request provides the area description again, which enables a rapid look-up from recent requests of offered variant objects, the variant names required are then listed. The data from the variant objects described is extracted and combined into a single response. The first variant for each request should be the variant describing the relevant tile header data, this is the only level of structure implemented in the current server (due to limitations in the NTF format, which describes road names with coordinates and does not establish clear links in data with some other potentially structured features). Variants from multiple tiles are requested in multiple (possibly parallel or pipelined) HTTP GETs.

Extracting typing encoded within data is also possible. NTF uses one of a set of well defined feature codes within a feature definition. Different data sets (Meridian, Strategi etc.) use different feature codes. We choose to allow mappings between multiple types in our definition and multiple types in the data's definition, for maximum flexibility. These relationships are defined in a simple XML format, an example of which is given below:

```
<type-mapping domain="map"
  foreignformat="ntf"
  foreignfield="feature-code">
  <map>
    <type name="building:rail-station"/>
    <type name="railway"/>
    <foreign name="6155"/>
    <foreign name="6551"/>
    <foreign name="5520"/>
  </map>
</type-mapping>
```

Here we see the mapping between the various feature codes for a railway station in NTF: 6155 (Meridian railway station), 6551 (Meridian station name), 5520 (Strategi railway station); and the type system definitions `building:rail-station` and `railway` in the type domain `map`.

## 5.2 Benefits vs. Overheads

It is important to consider the impact of retrieving the meta data on the system. The transfer of this data will take time, which has to be considered against the benefit of using it.

We examined twenty 10km\*10km map tiles for both urban and rural areas, measuring average data and meta data

volume. We found that while there is a clear overhead it is not prohibitive: 13% for urban data and 43% for rural data, which already presents less of a resource problem. In addition we find that the overhead presented is more stable (lower standard deviation) than the data itself. This is due to the grouping of features and the fact that a line feature in a map is encoded as a sequence of points, while its meta data is the same size as the data for a point feature. In many cases much of the data is omitted and the meta data uses less bandwidth than the selection saves, so provides a benefit overall. Having said this it would be desirable to reduce the meta data overhead. The language may be further compressed and future work may address data efficiency further.

Possibly of more concern than size is, that the meta data imposes an extra request / reply delay on the interaction. With many slow or highly loaded networks this could be seen as a serious problem: both in terms of responsiveness under good conditions and in terms of taking up time which might be better spent transferring data. We have two responses to these concerns: first, that the impact will often be small enough that the benefit of more appropriate application behaviour will outweigh the impact of the overhead; second, that if the last-hop network is the source of a large part of this extra delay, then a proxy with better connectivity should be considered to perform the mediation rather than the client.

Another alternative, explored in some prior work, e.g. [15, 13] is server-side adaptation of data, presenting the client with a tailored response. This requires either a detailed specification of requirements, or a description of the context – which then requires that the needs of the context can be inferred. Sending description of all requirements and preferences is not always an obvious choice: our specifications for mediating maps, for testing purposes, are 30kB and so an overhead in themselves; as resources vary and privacy issues arise clients may be unable to provide a fine grained description of their context, or unwilling to describe their requirements and/or context to unknown 3rd parties. In addition, requiring servers to maintain state for all clients is not necessarily desirable.

In tests we found the mediated maps to be more predictable in their download time than un-mediated maps. Useful maps were produced to 15-60s deadlines, which were shorter than loading all the map data over sub-56kb/s networks. Predictable delays have been shown to be a key factor in user satisfaction in the web. The omission of unnecessary detail was shown to provide improved clarity and faster navigation in user tests [3].

## 6 Conclusions

We have described a rich and flexible meta data model which integrates:

- Information and data structure through trees of *elements* and *variants*. This supports adaptation of all elements of a document which is consistent across elements and considers the total effects of the selection, so allowing effective resource constraints on the selection, e.g. deadlines.
- Data *variants* realising information *elements* in a flexible way, allowing variants to contain multiple elements – thus separating the logical information structure from its encoding.
- Semantic properties of elements are described through *types* and *extensions* such as bounding area for geographical features. These provide for context awareness by allowing the prioritisation of elements.
- Syntactic properties of variants are described through attribute descriptions such as data size. These are extensible according to mime type, e.g. `text/*` format data may have a word count, `image/jpeg` format data may have a compression level. This description aids context awareness by allowing the selection of the most appropriate data forms and controlling resource use, such as bandwidth.

In addition to the abstract model we have illustrated our XML encoding of these concepts through two example context mediated applications, one showing maps the other web pages. Supporting the meta data we have shown our multi-domain semantic type system, mappings between encoded semantics and our type system and commented on the practicalities of implementing the meta data in these applications. The approach is flexible: the XML syntax and type systems are easily exchanged for other techniques. The meta data model supports a very rich description of data to enable powerful contextual mediation of data – providing for more responsive and appropriate behaviour in the diverse world of mobile and ubiquitous computing.

There remains work to be undertaken in this area, including: encoding our model within standardised XML; application of more complete and complex ontologies to element typing; investigating compression and caching properties of the meta data to improve performance; and applying the meta data to support context aware presentation and informing users of the omission of data. We are currently examining the application of our data model and mediation techniques to CSCW and messaging style applications. The generic basis for our metadata and its use ensures that much of the code developed remains applicable across application domains.

## 6.1 Acknowledgements

This work was funded by the EPSRC platform grant GR/R 95715/01 “Aedus”. Map data were generously made available by Ordnance Survey.

## References

- [1] T. Baker. Languages for Dublin Core. *D-Lib Magazine*, December 1998. <http://dublincore.org>.
- [2] D. Bulterman. User-centered abstractions for adaptive hypermedia presentations. In *ACM Multimedia '98*, 1998.
- [3] D. Chalmers. *Contextual Mediation to Support Ubiquitous Computing*. PhD thesis, Imperial College, London, UK, 2002.
- [4] D. Chalmers, N. Dulay, and M. Sloman. A framework for contextual mediation in mobile and ubiquitous computing. *Personal and Ubiquitous Computing*. to appear.
- [5] O. G. Consortium. Geography Mark-up Language (GML) 2.0. Technical report, Open GIS Consortium, 2001. <http://www.opengis.net>.
- [6] A. Fox, S. Gribble, Y. Chawathe, and E. Brewer. Adapting to network and client variation using active proxies: Lessons and perspectives. *IEEE Personal Communications*, 1998.
- [7] K. Holtman and A. Mutz. Transparent content negotiation in http. RFC 2295, IETF, 1998.
- [8] M. Hori, G. Kondoh, K. Ono, S. Hirose, and S. Singhal. Annotation-based web content transcoding. In *9th International World Wide Web Conference (WWW9)*, 2002.
- [9] P. Hoschka. Synchronized Multimedia Integration Language (SMIL) 1.0 specification. Recommendation, W3C, 1998.
- [10] B. Kerhervé, A. Pons, G. v. Bochmann, and A. Hafid. Meta-data modelling for quality of service management in distributed multimedia systems. In *1st IEEE Metadata Conference*, 1996.
- [11] G. Klyne. A syntax for describing media feature sets. RFC 2533, IETF, 1999.
- [12] O. Lassila and R. Swick. Resource Description Framework (RDF) model and syntax specification. Recommendation, W3C, 1999.
- [13] R. Mohan, J. R. Smith, and C.-S. Li. Adapting multimedia internet content for universal access. *IEEE Transactions on Multimedia*, 1(1):104–114, 1999.
- [14] K. Nagao, Y. Shirai, and K. Squire. Semantic annotation and transcoding: Making web content more accessible. *IEEE Multimedia*, pages 69–81, April-June 2001.
- [15] F. Reynolds, J. Hjelm, S. Dawkins, and S. Singhal. Composite Capability / Preference Profiles (CC/PP): A user side framework for content negotiation. Note, W3C, 1999.
- [16] J. Smith, R. Mohan, and C. Li. Content-based transcoding of images in the internet. In *IEEE ICIP-98, Special session on Network-based image processing*, pages 7–11, 1998.
- [17] G. v. Bochmann, B. Kerhervé, A. Hafid, P. Dini, and A. Pons. Architectural design of adaptive distributed multimedia systems. In *International Workshop in Distributed Multimedia Systems Design*, 1996.
- [18] S. Wirag. Modelling of adaptable multimedia documents. In *European Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS 97)*, 1997.